

Feedback Comment

: "you already have a keys.js file, why are you including the twitter keys again in your liri.js files multiple times?",

: "So, it looks like you copied from Bilal? But forgot to copy the whole file structure? Let me know if you need any help to review this hw. ",

: "in your liri.js file , looks like you missed on to declare the keysJs variable and missed to require the files and modules that your need.",

: "for your switch statements you need to put that inside a function so you can call it when you want to as a whole instead of having it all open.",

: "instead of having the if statements floating outside, it's better to have them all in one place inside a function, and then call the function so it can run it when the function is called.",

: "also use proper indentation before pushing to github or use a beautify plugin",

: "Good job!",

: "i'm not sure why you have a another liri.js file which is incomplete and also why you're .gitignore is named incorrectly. Also, ",

: "why do you have your keys.js file inside your gitignore. The .gitignore file is used to add files that you want github to ignore.",

: "Good start but I notice you didn't do too much/finish. Did you have some trouble starting? ",

: "Good effort. Was the tweets function something you had trouble with or just ran out of time?",

: "good job on the hw, but always remember to DRY your code. For eg, in your spotThisSong() you're doing this
`console.log(data.tracks.items[0].artists[0].name) console.log(data.tracks.items[0].name)`
`console.log(data.tracks.items[0].preview_url) console.log(data.tracks.items[0].album.name)`'. Instead you could loop it and console.log it out instead of logging it out each time.",

: "Keep up the good work!",

: "Good start but it looks like you didn't finish. Did you have some trouble or did you run out of time?",

: "I love the introduction screen!!! great comments great indentations everything worked including the bonus!! lines 134-147 in your code you JSON.parse the body a bunch of times. I would do this once, store this into a variable. And then use that variable.",

: "I would make a package.json file :)",

: "Nice job! ",

: "Great Job! I like how you prompted for the twitter username too even though we didnt ask you to do that. Keep up the good work!",

: "Good effort!",

: "copied from david - don't do that again.",

: "hey Jeff, I would throw your node_modules and keys.js folder and file into the .gitignore file. Read the assignment instructions again. It would've been great if you made a package.json file.",

: "Hey Joel, looks like you got started here but gave up early. Hopefully you can find time to finish this at some point. Let me know if you want help going over it.",

: "Nice job getting this in early Davon!",

: "great job man. I would throw in a package.json file, but other than that I'm proud of your work :)",

: "I would jazz up your readme file. It'll get you used to writing markdown. \n\nI would also throw in a package.json file to make collaboration easier. \n\nI like how you updated your log file with a timestamp and the user input. \n\nTo fully complete the bonus, you'd have to make a log function and call the log function inside of twitter, spotify and movie functions, and additionally log the data that you console.log",
: "First, awesome job making your Read Me \"more prettyfull\", you definitely succeeded.\nYou probably realize this by now, but you dont need the \"package.json\" or \"index.php\" files now that we're using Node. But overall, great job! Your code is neat and efficient. The only thing I'd say you might change is generally you want to require all your files and packages within the first few lines of the file, regardless of where you're using them. But thats not a huge deal, mostly just convention, it shouldn't affect how the code works.\nP.S. I got rick rolled and Im not even mad about it. Nicely Done!",
: "Hey Ryan, great job! Your code is neat and well commented, and nice work including the bonus too! One minor thing you want to keep in mind is generally you want to require all of your files and packages at the top of the page, regardless of where they are used, for readability. And for something like \"var request = require(\"request\");\", if you include it once at the top of the page, you dont have to include it in each function. Keep up the good work!",
: "Hey Bilal, fantastic job! Your code is neat and well commented, and good job on the bonus too. Also, I really like how youre taking in a twitter handle so you can display anyones tweets instead of your own. One thing to keep in mind is that you didnt include your \"keys.js\" file in your \".gitignore\". It doesnt really matter for this project, but if you dont do that, and you upload your project to github, your private keys are available for everyone to see, just like your keys are now. So get those keys in your .gitignore and keep up the great work!",
: "To call the ombd API I would use require the \"request\" package & require it on the top. Would also define your process.argv[3]
: "Nice work getting this in early!",
: "awesome job on the application, like the \"for in\" loop you threw in your twitter function. also like the added colors package. great readability w/ easy to understand comments",
: "Looks like you had the right idea including all the required packages, would use process.argv[i]
: "Lines 1-109 are the same as lines 110-219, two sets of the same main app code. Like the effort I'm seeing with your twitter, spotify & request functions. I would go back and review process.arv[i]
: "Nice job getting this in so early William!",
: "awesome job the application - good call remembering to keep your keys hidden. like the initiative with your log action function",
: "Nice job getting this in so early! ",
: "great job on the application, one suggestion would be to include all your local variables var twit & var params within the scope of your twitter function",
: "Would put your \"new Twitter\" in your twitter function to export your keys - https://www.npmjs.com/package/twitter Please work on syntax/readability (indentation, removing extraneous comments Good job making the request to ombd, there is no need to require an \"ombd\" package. Good overall improvement, I see use of switch cases!",
: "Good job on deleting your keys.js file: https://github.com/angelica3010/liri-node-app/commit/0d94e5324b2bb0eb24425c1cb7fad4808e5ef35d \n\nYou'll have to run some commands to get permanently get rid of keys.js from your git history\n\nInstead, you can delete the repo, recreate it and then push up your code to it.",

: "Hey Frankie, I cloned your repo and tried to run your app.\n\nWe talked about this in class a few weeks ago. But looks like the data for this feedback got lost. Looking through your `l.js`, I see a constant switch between camelCase and under_score variable naming, which essentially is breaking your app. I'm getting a lot of `undefined`s, because you're calling variables one name and trying to use them by another name. \n\nAlso, the `Twitter_Get` fn doesn't have consistent naming for variables e.g. `var par` & `client.get('statuses/user_timeline', params. Also the screen_name key should have a value of a screen name besides user.\n\nIn your `Spotify_Get` and `Movie_Get`, you've changed the default values, which were specified in the homework instructions. I just don't see a reason for this. While it shouldn't matter, the function don't work also again because of variable naming inconsistency. \n\nFrom this week's homework (I know it was a while ago), you've got to stick to a naming convention and test your app as you write functions and again before you're finished. While there isn't a standard convention on variable naming for JavaScript, you should work with the convention your `team` in this case the course works in, which is camelCase.",

: "Also, I added a controller.js in the root menu. :-D it's in the modularize branch.",

: "My program was in working order by the deadline -- my README file however did not have all the screenshots in time for the deadline. But hey it totes works. :D",

: "having issues with GitHub for close to a week. Last week I created new repos and I was able to submit. This week I create new repos and I get a fatal error that repos already exists. Then I try to push files and receive message that repo does not exist. ",

: "Forgot to comment. The Videos had to be <50mb so split them into the repo. You'll see the avi files and you may have to pause the file to see the inventory updates via sql.",

: "good job. ",

: "this was missing package.json. From the screenshots, looks all right. ",

: "good job. Though there's no screenshots.. which I'm pretty sure it was suppose to be included. ",

: "Thank you for the README. Looks great. ",

: "I was able to get spotify & twitter to work. \n\nI will work on rotten tomatoes if I get time.",

: "forgot to submit on Saturday because I was not in class.",

: "Got help to get the whole thing done because I'm not that comfortable enough with pure JavaScript to do it on my own. Also changed the default song to `Ripple` by The Greatful Dead because whoever wrote the instructions has no idea what real music is.",

: "I'm glad you attempted it and worked on it again! Always refer to the activities class and let me know if you want some 1 on 1 help.",

: "I like these shortcuts! `if ((launchPhrase == 's') || (launchPhrase == 'spotify') || (launchPhrase == 'spotify-this-song'))\n{spotify(songSearch);}nelse `n `if((launchPhrase == 't') || (launchPhrase == 'twitter') || (launchPhrase == 'my-tweets'))\n{twitter(twitterSearch);}n `else `n `if((launchPhrase == 'm') || (launchPhrase == 'omdb') || (launchPhrase == 'movie') || (launchPhrase == 'movie-this'))\n {movie(movieSearch);}n `else\n {doThis();}n`",

: "Great job! one piece of advice:\n\nvar omdb = require('omdb');\n\nPut your require statements at the top of the code. It's a lot easier to see all the dependencies at once. ",

: "Good job starting this homework. Try to work on the other apis! \n\nOne piece of advice, don't loop an array using a hardcoded number. instead use the array's length property:\n\nif (task == 'my-tweets') {\nvar params = {screen_name: 'smgalante', count: 20};\nclient.get('statuses/user_timeline', params, function(error, tweets, response){\nif(!error){\nfor (i = 0; i < 20; i++)\n{\nconsole.log(tweets[i].text);\n}\n}\n}\n}\n\nthis section, use tweets.length instead of 20. Because this will only do the first 20. But if there's less/more you will get an error. tweets.length guarantees you to iterate through each element in the tweets array",

: "Awesome job. No area that requires improvement here. On your readme file, start writing instructions about the app and how to install it. something like\n\nInstructions:\nnpm install\nnode liri",

[illegible]

: "Travis, do you think you can work on this now? Let's schedule a time and work on it together, I think you can do it with a little bit of time",

```
: "Great job. This is really important:\n\n\\n// Search for a movie\nfunction getMovie () {\nvar omdb =\n'http://www.omdbapi.com/?t=';\nvar omdbTomatoe = '&y=&plot=short&r=json&tomatoes=true';\nvar omdbUrl = omdb + movie +\nomdbTomatoe;\n request(omdbUrl, function (error, response, body) {\n if (!error && response.statusCode == 200) {\n var\n results = \n      \"Title: \" + JSON.parse(body)[\"Title\"] + \"\\n\\n\" + \n      \"Rating: \" + JSON.parse(body)[\"Rated\"] + \"\\n\\n\" + \n      \"Country: \" + JSON.parse(body)[\"Country\"] + \"\\n\\n\" + \n      \"Language: \" + JSON.parse(body)[\"Language\"] + \"\\n\\n\" + \n      \"Plot: \" + JSON.parse(body)[\"Plot\"] + \"\\n\\n\" + \n      \"Actors: \" + JSON.parse(body)[\"Actors\"] + \"\\n\\n\" + \n      \"Rotten Tomatoes Rating: \" + JSON.parse(body)[\"tomatoRating\"] + \"\\n\\n\" + \n      \"Rotten Tomato URL: \" + \n      JSON.parse(body)[\"tomatoURL\"] + \"\\n\\n\";\n console.log(results);\n } }\n});\n\nJSON.parse is a heavy calculation.\nInstead of doing this on every property, do JSON.parse(body) once and save it to a variable\n\nvar jsonBody =\nJSON.parse(body);\n\nthen you can access all the attributes using dot notation:\n\njsonBody.Title\njsonBody.Year"
```

: "Good job, only thing left to do here is the reading of random.txt so that it will run the command in the file. There's a few students taking this series of videos:\n<https://www.youtube.com/watch?v=yQaAGmHnN9s&list=PL46F0A159EC02DF82>\n\nI recommend you code along with this (not just watching), it will be really helpful. ",

```
: "JSON.parse is a heavy computation. Instead of doing it for every attribute, save the json body once:\n\nvar jsonBody =  
JSON.parse(body);\n\nthen you can do jsonBody.Title, jsonBody.Year, and so on.\n\nvar movieFunction = function() {\n  if  
(argv3 == undefined) {argv3 = "Mr. Nobody"};\n  var query_url = 'http://www.omdbapi.com/?t=' + argv3 +  
&y=&plot=short&r=json';\n  request(query_url, function(error, response, body) {\n    if (!error) {\n      console.log("Title: " +  
JSON.parse(body).Title + "\n      Year: " + JSON.parse(body).Year + "\n      IMDB Rating: " + JSON.parse(body).imdbRating  
+ "\n      Country: " + JSON.parse(body).Country + "\n      Language: " + JSON.parse(body).Language + "\n      Plot: " +  
JSON.parse(body).Plot + "\n      Actors: " + JSON.parse(body).Actors + "\n      Metascore: " + JSON.parse(body).Metascore  
+ "\n      Country: " + JSON.parse(body).Country + "\n    ");\n  })\n}
```

```
: "Good job! Only thing left to do is to do the 'do-what-it-says' part. Another recommendation, put your dependencies up at the top:\n\n var Twitter = require('twitter');\n var keys = require('./keys.js');\n\nThat way, developers can see all the dependencies at the top of the page. ",
```

: "I need to go back and do all the api stuff. This is incomplete."

: "Amanda I'm going to wait on this homework, please try your best, I know you can do it!",

: "Good job on your homework. You were on the right path. Try to get back to it and finish the `movieThis()` and the `whatItSays()` functions when you get a chance. :)",

: "Susan, make sure to take a look at the solution to see the steps you were missing. Here are some tips I have. After installing the packages with npm install, make sure to 'import' those dependencies into your file by 'requiring' it on the top of your file. Then, my suggestion is to take a look at the npm module documentation to see how you can actually get the data. For example, if you go to the npm module for twitter, you will see a method that will allow you get the tweets. And there you will see that you'll need the keys and where to pass them on your call. Let me know if you need help to understand the solution. :)",

: "Sheli, you were in the right direction with a few bugs. First, spotify was misspelled on the top of the page. :) Then, you were requiring the random.txt file, which isn't needed and was breaking your program for some reason. Some other things to point out are, your Tweeter keys were already in your keys.js file. Take a look at the solution to see how you could have used them inside your liri.js file. And on your spotify function, you were not passing the parameter with the name of the music to your function. If you do so, you can then pass that into your search object to search for that music. The same should be done with your function movie(). You need to pass the userInput to that function and call it. Let me know if you need help understanding those things. They are mainly related to how functions work and I'll be glad to help you. :) ",

```
: "Great job on your homework! And thank you for using a package.json file. It made my life easier :)",
```

: "Great job on your homework. The only thing I have to point out is that your keys.js is on your Github Repository. You probably added it to the gitignore after you had already pushed it to Github. One way to fix it is by removing that from your directory (it will potentially break your code momentarily), then commit the changes, then move it back and commit the changes again. Let me know if you need any help with that. Oh! One more thing! Instead of creating a var value inside your function, you could have passed it as a parameter to your spotify and movie functions. :)",

```
: "added inquirer.prompt()\nremoved process.argv based commands",
```

: "Awesome work. I like that you are starting to commit more often and with more detail.",

: "Great job with this assignment, Caleb! I don't have a ton of feedback as you did a great job! Keep up the good work!",

: "Great job with this Crystal! I especially liked how you formatted your responses to be easier to read in the console. Everything works great and your code is very clean! My only slight suggestion would be to put the process.argv calls into a variable to increase readability but overall it was great!",

- **III**
- **,**

```
: "Great job with this Danny! I love to see the improvement you've made over the last few weeks! The only command I couldn't get
to work is the 'Do What It Says' one. You have all of the other commands working great, just make sure you're reading the file
contents and calling the proper function. If you're having trouble with this, come see me."
```

: "Good job Pinar and thank you for making a package.json. It's actually very helpful for the other developers who will see your code. :) One suggestion I have is to look at passing the `process.argv[3]`, into your `random` function and then pass that to your `movie` and `spotify` functions when you call them inside the switch statement. Just make sure to pass a parameter to those functions when you create them. Let me know if you need more clarification in what I'm trying to say. ",

: "Great work! I'd consider cleaning up your syntax a bit by using more consistent spacing and indentations.\n\nConsider breaking up [long lines](https://github.com/mgusmano84/Liri-Node-App/blob/master/liri.js#L44) into separate variables (so they're easier to read):\n```\nvar content = '\nSpotify' + '\nArtist: ' + data.tracks.items[0].artists[0].name + ' \nSong Name: ' + data.tracks.items[0].name + ' \nPreview Link: ' + data.tracks.items[0].preview_url + ' \nAlbum: ' + data.tracks.items[0].album.name + ' \n\n*****',\n nfs.appendFile('.log.txt', content, function(err) {\n \n You could also take this a step further and use [Array.join()](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Array/join) rather than concatenating long strings:\n \n var content = [\n 'Artist: ' + data.tracks.items[0].artists[0].name,\n '\n 'Song Name: ' + data.tracks.items[0].name,\n '\n];\n \n nfs.appendFile('.log.txt', content.join('\n'), function(err) {\n \n \n });\n \n Also, instead of [invoking `JSON.parse()` multiple times](https://github.com/mgusmano84/Liri-Node-App/blob/master/liri.js#L83-L89), I'd only use it once to save some processing time and keep your code cleaner:\n \n var data = JSON.parse(body);\n \n console.log(`Title: ` + data.Title);\n \n console.log(`Released: ` + data.Year);\n \n console.log(`IMDB Rating: ` + data.imdbRating);\n \n ``,

: "Great job with this assignment Daniel! I especially like how you formatted the console logs to make it easier to read! Not much feedback as you did a great job!",

<p>: "Make sure to add your `keys.js` to [`.gitignore`](https://git-scm.com/docs/gitignore) so that it does not get pushed to github (otherwise, someone malicious could use your twitter account)! Good use of `package.json` but make sure to include your `node_modules` folder to `.gitignore` I would consider moving the logic within your switch statements, into separate functions (large chunks of code between `case`, can make them difficult to discern) - similar to your `movie-this` and `spotify-this` case. Instead of repeating the same logic in your `do-what-it-says` case, I'd put my first `switch` into a function and call it. Great work!</p>
<p>: "Good job Patrick. The one suggestion I have is to look at passing the process.argv[3], into your random function. Then pass that to your movie and spotify functions when you call them inside the switch statement. Just make sure to pass a parameter to those functions when you create them. Let me know if you need more clarification in what I'm trying to say."</p>
<p>: "You were close with this assignment, Dawn but you had a lot of small mistakes that held you back. Mainly with how the functions were created, syntax errors, and the use of process.argv. I have gotten most of it to work with this code: <pre> var fs = require('fs'); var keys = require('./keys.js'); var Twitter = require('twitter'); var spotify = require('spotify'); var request = require('request'); var params = process.argv[2]; var userInput = process.argv[3]; var pick = function() { switch(params) { case 'my-tweets': twitterStatus(); break; case 'spotify-this-song': if(params[1]) { spotifySong(userInput); } else { params[1] </pre> </p>
<p>: "Great job with this assignment! Looks like the only one I couldn't get to work is the do-what-is-says call. You've got all of the pieces in place so just make sure you are reading the file with fs and then calling the function you've created with the arguments you pull out of the document. Great job though!",</p>
<p>: "Good job Nigel. The one suggestion I have is to look at passing the process.argv[3], into your random function. Then pass that to your movie and spotify functions when you call them inside the switch statement. Just make sure to pass a parameter to those functions when you create them. Let me know if you need more clarification in what I'm trying to say."</p>
<p>: "Good job with this assignment. Just be sure you are following the instructions for the commands. It's a small feedback but you'll get clients that want things a certain way when you get out in the real world and you need to be sure you follow the instructions for how commands are executed. Namely talking about `get-song` vs. `spotify-this-song`. Also, I wasn't able to get the `do-what-it-says` call to work as you didn't define the function to do that one. You have all of the pieces in place just make sure you are reading the file using `fs` and then calling the proper function with the work you've already done. Almost there but good job for the most part!",</p>
<p>: "Max, please refer to the homework solution and let me know if you would like me to go over it with you. I'd be glad to help."</p>
<p>: "I'm not sure what happened here but I didn't see any code in the repository. Come see us if you're having trouble understanding these concepts."</p>
<p>: "Great job with this assignment! Everything works as designed and your code looks nice and clean! I like how you formatted the console.logs as well as it made it easy to read and test. Good job!",</p>
<p>: "Good use of `JSON.stringify()` to store text in your `movie` function but, it's missing from your `song` function. I'd also consider wrapping your switch statement within a function, that way it can be called for the `do-what-it-says` case (as it stands, only the `song` method will work). Overall, great work!",</p>
<p>: "Make sure to include `keys.js` and `node_modules` in `.gitignore` (https://git-scm.com/docs/gitignore) It looks like you have a [getArtistNames()](https://github.com/PoffDev/nodeJS-Homework/blob/master/liri.js#L75-L80) method that is never used. Try to keep your code consistent via line breaks and indentations. The `runThis` function isn't needed since you can just invoke `pick`. Overall, _great_ work! ([this](https://github.com/PoffDev/nodeJS-Homework/blob/master/liri.js#L32) was a great touch)",</p>
<p>: "Great use of `package.json` Try to keep your code consistent via line breaks and indentation. I'd consider using a local variable (rather than global) for `logData` and pass information as an argument to `writeLog()`. <pre> function writeLog(data) { //reads text files and returns contents to data fs.appendFile('log.txt', JSON.stringify(data, null, '\t'), (err) => { if (err) { console.log('Error occurred: ' + err); return; } }); writeData({ something: 'meow meow meow' }); } </pre> Good use of `qs` (query string), for your request Overall, great work!",</p>

: "Make sure to include `keys.js` into `.gitignore`. \n\nInstead of duplicated switch statements within your `doSomethingRandom()` method, I'd consider breaking those cases out into a function and calling that.\n\nOverall, great work!",
: "Good job, from everything I can see you did everything correctly. However make sure that you submit all required files in order for us to test/run your program. I was missing your keys file. Keep up the good work.",
: "Great job writing your liri program. Make sure you comment your code and we needed you to submit a json file to support your required npm files. Keep up the good work.",
: "Good job on writing your code, its becoming much cleaner than before. One thing I would like you to work on is commenting. Not only will it help you to understand and comprehend better in your code it will help others to read your code easier. Make sure you always submit the required files, we were missing the keys file as well as the json to install the npm modules. Keep up the good work.",
: "Great job, keep up the good work.",
: "Great job, clean code, nice comments.",
: "Make sure when you submit your homework that all required files are submitted. We are missing the keys file as well as the json packages file for the node modules. Try to comment your code more, not only will this help you in understanding what you are writing, but if you ever have to go back to your code later you can figure out what in the world happened. You fellow team members will also appreciate that.",
: "Great job, keep up the good work. I really appreciate everything you have been doing to assist your fellow classmates. :)",
: "Nice work. It didn't work initially for me because the package.json did not have spotify as a dependency.",
: "Nice work I like the colors added to the outputs. In the future don't push keys.js to git. That's your private key. You need to add it to your .gitignore before you push to keys.js",
: "This looks good. Shouldn't have pushed keys.js that's your private key. Homework stated to add it to your .gitignore. Also, you didn't push the package.json file to git. This is needed so we can get your apps dependencies when we run "npm install". Overall great job!",
: "Come see myself or Victor before/after class. Also, remember you can contact Ricky to setup 1 on 1. You'll get it!",
: "You shouldn't have a file named node_modules when trying to install dependencies from your package.json it was throwing an error because of that file. \n\nShouldn't have pushed keys.js that's your private key. Homework stated to add it to your .gitignore it looks like you did but it was after you had already pushed keys.js\n\nOverall Excellent job! The formatting of the returned data was great!",
: "This is awesome! You didn't push the package.json file to git. This is needed so we can get your apps dependencies when we run "npm install""",